Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 585634, 6 pages
http://dx.doi.org/10.1155/2014/585634

Research Article

Chinese Location Word Recognition Using Service Context Information for Location-Based Service

Jiujun Cheng,[1,2] Jingyu Hou,[2] Chendan Yan,[2] Junlu Cheng,[2] and Qingyang Zhang[2]

[1] *Department of Computer Science and Technology, Tongji University, Shanghai 201804, China*
[2] *Key Laboratory of Embedded System and Service Computing of Ministry of Education, Tongji University, Shanghai 201804, China*

Correspondence should be addressed to Jingyu Hou; houjingyu_0321@hotmail.com

Received 18 December 2013; Accepted 8 January 2014; Published 13 March 2014

Academic Editor: Weichao Sun

With the development of mobile networks and positioning technology, extensive attention focuses on the location-based service (LBS) which processes the application data including user queries, information searches, and user comments by the location information. In LBS, the recognition of the location word in user messages is meaningful and important. The location word recognition in LBS is different from the traditional named entity recognition, owing to the additional information such as user location coordinates in LBS. This paper proposes a method that adds the service context information including user location coordinates and message timestamps into the machine learning to improve the accuracy of the Chinese location word recognition. The experiment based on microblog datasets in mobile environment proves the viability and effectiveness of this method.

1. Introduction

The rapid development of mobile networks extends the way for users to obtain information. One user can get information from network services and mobile applications without time and physical location restrictions. Mobile phones, PDAs, and other devices equipped with positioning technology expand the scope and content of the network services. Location-based service (LBS) [1] is gradually developed in such a situation. This paper is based on this background.

When a user makes a request in LBS, his position coordinate is provided to server as an input to the query processing. By content of service, the LBS usually contains navigation services, location-based query services, and location-based games. And according to the source of query results, the LBS is divided into concentrated service in which resulting contents are given by service providers and crowdsourcing one whose result providers are users.

In LBS, the location words constantly appearing are usually associated with the service content. Through the messages with these location words, server applications can recognize the user requirement and react accordingly. So recognizing these words has great significance to result processing and users targeting. For example, in a location-based query service, a query of a user is "is there a gas station right near Shanghai Automobile City" After this query is uploaded to a server, the server can recognize the word "Shanghai Automobile City?", and then push the message to users around Shanghai Automobile City who may answer this query. What is more, in microblog applications, location words can be useful for user's habits and behavior mining, and then corresponding recommendation information can be provided.

Location word recognition is one of named entity recognition (NER) tasks. The researches of NER mainly focus on the traditional text such as news articles. Related researches are mainly based on statistical methods or a combination of rule and statistical methods. Chen et al. [2] use two conditional probabilistic models including conditional random fields and maximum entropy models. Wu et al. [3] propose a hybrid Chinese named entity recognition model based on multiple features. Sobhana et al. [4] describe a system for Geological text that makes use of the different contextual information of the words along with the variety of features that is helpful in predicting the various named entity classes.

Currently specialized researches and training datasets for NER in LBS applications are relatively scarce. A similar research field, named entity recognition of the microblog in

mobile environments, has made some progress. Qiu et al. [5] conduct an experiment to show the distinction between microblogs and traditional news articles. Liu and Zhou [6] propose a system by conducting two-stage NER for multiple similar tweets. All these studies merely consider the relationship between words in text content, which has no essential difference from traditional researches. In the scenario of LBS, besides the text content, additional service context information can be used in location word recognition, which is not involved in other researches.

In this paper, we present a novel method which adds service context information including user location coordinates and message timestamps to improve the accuracy of the location word recognition and provide schemes that transform the information to features. Finally, relative experiments prove the viability and the effectiveness of this method.

2. Conditional Random Fields

Some machine learning models including hidden Markov model (HMM) [7], maximum entropy Markov model (MEMM) [8], conditional random fields (CRFs), and support vector machines (SVMs) [9] are suitable for the location word recognition. Compared with other models, CRFs model shows higher precision and is widely used recently. In our research, we adopt CRF as the training model as well.

2.1. Training Model. Lafferty et al. presented the conditional random fields (CRFs) in their paper [10] in 2001. CRFs model is an undirected graphical model that encodes a conditional probability distribution by a given set of features.

Given an observation sequence $X = (x_1, x_2, \ldots, x_m)$, the conditional probability of the state sequence $Y = (y_1, y_2, \ldots, y_m)$ is

$$P(Y \mid X) = \frac{1}{Z(X)} \exp\left(\sum_{t=1}^{m}\sum_{i}\lambda_i t_i(y_{t-1}, y_t, x, t) + \sum_{t=1}^{m}\sum_{i}\mu_i s_i(y_t, x, t)\right), \tag{1}$$

where $Z(X)$ is defined as the normalization factor. Let t_i and s_i, respectively, denote transfer feature functions and state feature functions. And λ_i and μ_i are defined as parameters of corresponding feature functions, which can be computed by parameter estimation algorithms, such as stochastic gradient descent (SGD) [11], averaged-perceptron (AP) [12] and L-BFGS [13].

2.2. Feature Functions. In CRFs, the features of the training set are expressed by feature functions. Feature functions are two-value functions that are extracted from the training set. Feature functions are defined as $f_i(y_{t-1}, y_t, x, t)$. The set of the real feature $b(x, i)$ should be known before defining the feature functions whose values are $b(x, i)$ when y_{i-1} and y_i are given:

$$f(y_{i-1}, y_i, x, i) = \begin{cases} b(x, i) & \text{if } y_{i-1} = B, y_i = I, \\ 0 & \text{otherwise,} \end{cases} \tag{2}$$

where $b(x, i)$ is the real observation function whose value is 1 when some real observation appears and 0 if not:

$$b(x, i) = \begin{cases} 1 & \text{the observation in position } i \text{ is "some",} \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

Thus, the corresponding feature functions can be acquired via obtaining $b(x, i)$, the observation in position i.

3. Location Word Recognition Using Service Context Information

The works of the traditional named entity recognition use the words or characters themselves and the relationship between words or characters as the features. In LBS, the available information is not only the text information, but also the service context information such as the user location and message timestamps when the users use the service.

Definition 1 (information by users). It describes the information when a user sends a message or a query in LBS:

$$\text{Info} = \left(c_i, l_{u,t_i}, t_i\right), \quad i \in I,\ u \in U, \tag{4}$$

where u is the user ID and i is the information ID. c_i is the word context in Information i; it can be gained by text input or speech-to-text input. t_i is the time when the messages or queries are created. l_{u,t_i} is the user location coordinate when the user u sends a message at time t_i.

Our task is to detect location words in c_i. Location words do not merely cover geographical words but include some organization names and building names. Whether an organization name or a building name means a location word depends on its text context, which makes the recognition difficult. For example, *Tongji University* is an organization name. When it describes the campus, it is a location word. However when it describes the university system, it is not a location word.

3.1. The Impact of User Location Coordinates. In LBS, messages or queries provided by users are usually related to their locations. For example, when a user sends a message about flows of some subway stations, the most possible place where he is located is on the way to the subway station or near the subway station. From the Chinese character level, when appearing with the location coordinates in these places, the characters in the name of the subway station are more likely to be labeled as the characters in the location word. Because these location coordinates can be uploaded by users in LBS, we can use them as the features to improve the accuracy of the location word recognition.

As mentioned earlier, we can use the user location as a feature and add it to the feature template. Generally speaking, the location information gained from the user mobile phone contains the longitude and latitude by two float numbers. However, it is difficult to adopt these coordinates as the features directly. There are two main reasons.

(1) The coordinates are too precise, making the number of samples in each location feature too small. In extreme cases, there is only one sample in some location features.

(2) They can render the quantity of new features to become enormous.

Definition 2 (user location coordinates feature). It describes the feature using the user location coordinates

$$D = (X_m, Y_n), \tag{5}$$

where X_m is the latitude identifier of the feature area. Y_n is the longitude identifier of the feature area.

The location coordinate of a user is

$$l = (x, y), \tag{6}$$

where x is latitude coordinate of the user location, y is the longitude coordinate of the user location.

When the user location is contained by the feature area D, $l(x, y) \in D(X_m, Y_n)$, if $x \in [X_m, X_{m+1})$ and $y \in [Y_n, Y_{n+1})$.

The location coordinates of users are transformed to the features for the feature template. A map is divided into a number of areas for defining the user location features. We do not consider the administrative division. Every feature area is defined as a square, and the feature area itself can be considered as a set of user locations. Generally speaking, the approximate distance that separates each degree of latitude is 111 km. That is, the approximate distance that separates 0.01 degree of latitude is 1 km. The distance that separates each degree of longitude is different on different latitude. However the distance can be treated to be identical on the local scale such as a city. It is about 95 km in Shanghai on N31. So, the approximate distance that separates 0.01 degree of longitude is 1 km. We use a square whose side length is 1 km as a feature in this paper. The label describing a feature is a tuple $D(X_m, Y_n)$. The X_m and Y_n are the coordinates of the bottom left corner of the square whose digit is 0 from three digits after the decimal point. If the user location is located in a feature area, the tuple is (X_m, Y_n). For example, if the coordinates are 31.199248, 121.444993, the tuple is 31.19, 121.44. In Figure 1, the user location statistical result with Xin Zha Road Subway Station is shown. Every point represents the user location of a microblog.

The tuple cannot be directly adopted by the feature template. We establish the relationship between the tuple and the character and adopt the context window within previous two and next two characters for unigram features. The bigram features are Character (0)/Character (1) and Character (−1)/Character (0). So the relationship between the tuple and the character adopts these rules as well. For example, one of feature functions is

$$f(y_i, w, L, i) = \begin{cases} 1, & y_i = B \wedge w = \text{“shang”} \wedge L \\ & \quad = (31.19, 121.44), \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

The feature template is described by Table 1.

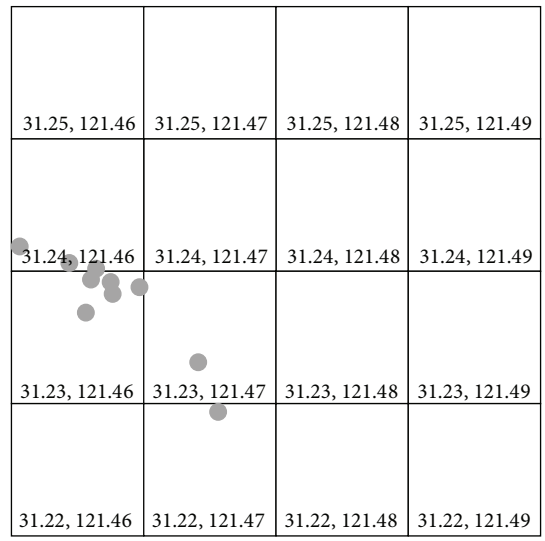

Figure 1: User location statistical result with Xin Zha road subway station.

Table 1: User location coordinates feature template.

Location and unigram	Location (0)/Character (−2)
	Location (0)/Character (−1)
	Location (0)/Character (0)
	Location (0)/Character (1)
	Location (0)/Character (2)
Location and bigram	Location (0)/Character (0)/Character (1)
	Location (0)/Character (−1)/Character (0)

3.2. The Impact of Message Timestamps. Users are concerned with different locations at different times. For example, in the morning and the evening peak, users are more concerned with the traffic information about the route they go, and messages or queries about department stores and markets increase at weekends.

Definition 3 (message timestamp feature). It describes the feature using the message timestamp

$$T = (w, h), \quad w \in W, \ h \in H, \tag{8}$$

where w is the type of the date in a week, W = {workday, weekend}. h is the type of the timestamp in a day, H = {0:00–5:59 & 21:00–23:59, 6:00–8:59, 9:00–11:59, 12:00–14:59, 15:00–17:59, 18:00–20:59}.

We use a week as a cycle. Meanwhile the weekday and the weekend are handled separately. For a day, the morning

Table 2: Message timestamp feature template.

Timestamp and unigram	Timestamp (0)/Character (−2)
	Timestamp (0)/Character (−1)
	Timestamp (0)/Character (0)
	Timestamp (0)/Character (1)
	Timestamp (0)/Character (2)
Timestamp and bigram	Timestamp (0)/Character (0)/Character (1)
	Timestamp (0)/Character (-1)/Character (0)

and the evening peak are, respectively, defined as three hours, and the time interval is defined as four hours at other times. Furthermore, the night and the dawn are considered as one timestamp feature. Each time interval is identified by a number.

Likewise, we establish the relationship between tuples and characters. The feature template is described by Table 2.

4. Results and Analysis

4.1. Experiment Setup. In order to prove the impact of user location coordinates and message timestamps on the location word recognition, we have done a comparative experiment. The open source tool of CRF we used is CRFSuite [14]. There are no available benchmark datasets for our environment. Therefore we use the datasets that were collected by ourselves. The datasets contain microblogs collected from the service such as the Microblog with location coordinates in mobile networks. These microblogs involve a location of interest to users. The range of the collected location is limited to Shanghai.

By the web spider and the search function of the Sina Microblog and Tencent Microblog, we have collected about 20000 original microblogs. However some useless microblogs need to be filtered out. Finally we reserve about 5000 microblogs with user location coordinates and timestamps. These microblogs need be processed in the preparatory work.

When a microblog includes the user location coordinates, we can see the label like http://place.weibo.com/imgmap/center=31.199248, %20121.444993.

The user location coordinates are the two numbers separated by the comma after the symbol =. Generally the former is the latitude and the latter is the longitude. The timestamp information can be gained after the label <*!* ⋯ >* *<*span class*="*ct*">. The formats of message timestamps are not unified in microblogs. So, we need to unify them before the experiment.

The microblog samples are shown in Table 3. The text context is the microblog published in Chinese by users. The location word is the word between the symbol "/". Under the microblogs, the English meanings of them are shown. The User locations are described by geographic coordinates. Under the coordinates, the tuples indicate feature areas of the locations.

Considering the unexpected effects of the Chinese word segmentation, we adopt the character-based tagging method.

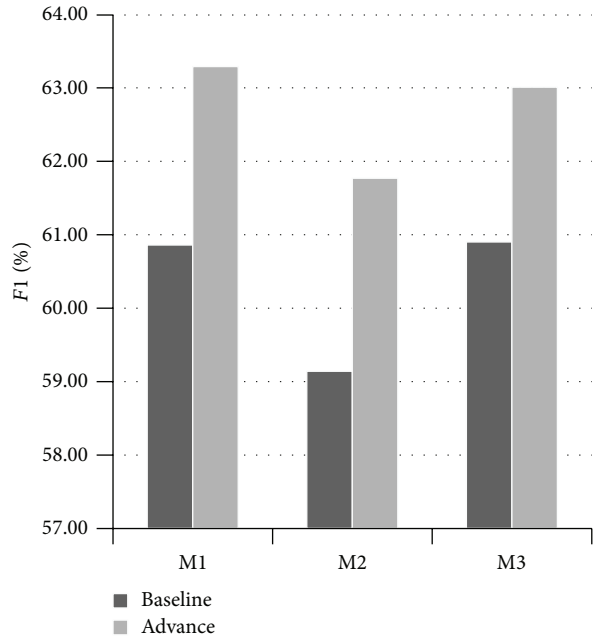

Figure 2: The performance with user location feature.

The labeling scheme we adopt is 3-Tag (B, I, O) [15]. The other features contain unigram and bigram features. The baseline feature template is given in Table 4.

Character is the Chinese character itself in the position. We adopt the template above in the baseline set in experiments.

For evaluation metrics, Precision (P), Recall (R), and $F1$ [6] are used. The main one is $F1$:

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \times 100\%. \quad (9)$$

The experiment contains two parts: (1) user location coordinates feature and (2) message timestamp feature.

From the dataset of 5000 microblogs, we randomly use 4000 ones as the training set and 1000 ones as the test set. We, respectively, implement this process three times on the dataset and the three newly obtained experiment datasets are named by M1, M2, and M3.

The training sets of each experiment datasets are trained three times, respectively, using baseline template, baseline with user location template, and baseline with message timestamp template as the feature template.

4.2. Results and Analysis. Part 1 is the comparison of the baseline and the baseline with user location. The comparison of the results measured by $F1$ is shown in Figure 2.

As shown in Figure 2, the accuracy of the recognition is improved after adding the user locations as the features. The $F1$ of the baseline in M1, M2, and M3 are 60.86%, 59.14%, and 60.90%. The $F1$ of the baseline with user location in M1 M2 and M3 are 63.29%, 61.77%, and 63.01%. The improvements of $F1$ in three environment groups are, respectively, 2.43%, 2.63%, and 2.11%. It is about 2%~3% in our experiment.

Table 3: Microblog data samples.

Text content	User location	Timestamp
我在/肇嘉浜路东安路/的公交站:我又迷路了, 路又不认识的, 电话也打不通, 怎么走啊?! (I'm at the bus stop of /Zhao Jia Bang Road Dong An Road/. I get lost, and can't get through. How will I do?!)	31.2004726491 121.453019886 (31.20, 121.45)	2013-03-06 20:10
/肇嘉浜路卢湾体育馆/不到点的上街沿, 夜排挡摊头开业了 (The night market has already begun near the /Zhao Jia Bang Road Lu Wan Gymnasium/.)	31.2062517893 121.468295738 (31.20, 121, 46)	2013-06-16 20:42
零次元校园行今天在/同济大学嘉定校区/, 校园很漂亮哟~期待今天中午的活动~ (Ling Ci Yuan Campus Activity will be held in /Tongji University/ today. The campus is very beautiful~ Looking forward to this event~)	31.28277 121.21611 (31.28, 121, 21)	2013-09-24 10:01

Table 4: Baseline feature template.

Unigram	Character (−2)
	Character (−1)
	Character (0)
	Character (1)
	Character (2)
Bigram	Character (0)/Character (1)
	Character (−1)/Character (0)

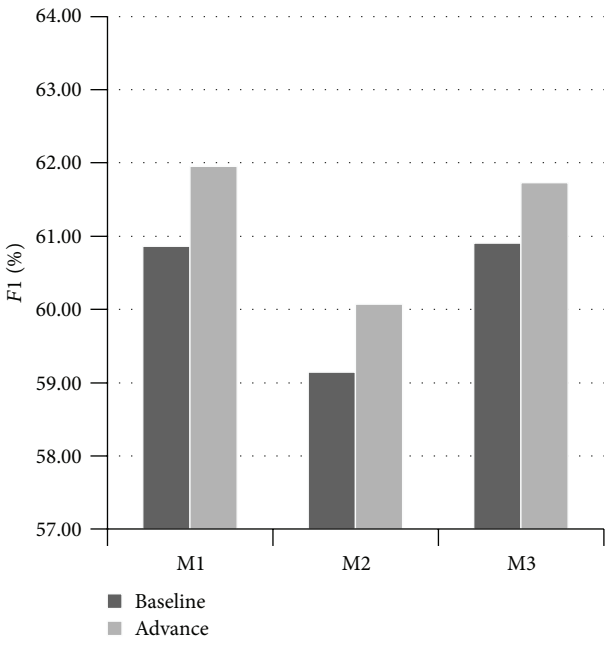

Figure 3: The performance with message timestamps.

Part 2 is the comparison of the baseline and the baseline with message timestamp. The $F1$ of the baseline with message timestamp in M1, M2, and M3 are 61.95%, 60.07%, and 61.73%. The comparison of the results measured by $F1$ is shown in Figure 3.

As shown in Figure 3, the performance is improved about 1%. The improvements are, respectively, 1.09%, 0.93%, and 0.83%. It is less than that in Part 1. The main reason is that the message timestamps are irregularly distributed

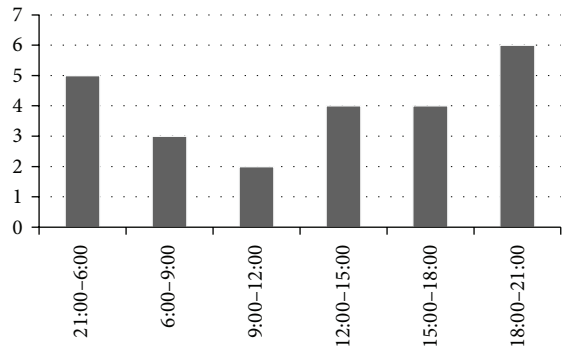

Figure 4: Message timestamps statistical result with Xin Long Road.

for some locations. For example, in Figure 4, it shows the message timestamp statistical result of a location named Xin Long Road. The distribution is very even and the feature is ineffective in this situation.

On the whole, the user location feature and the message timestamp one authentically improve the performance of the location word recognition in LBS. To some degree, it can correct the noise in the training set and improve the labeling probability of some characters. Furthermore, a case not noticed above is that some similar location words are usually contained by the same area. So the user location feature can improve the performance of the recognition for these similar words.

5. Conclusion

In this paper, we propose a method that adds the service context including user location coordinates and message timestamps as the features in LBS. The method gains acceptable performance in our experiment. The performance of the recognition with the user location coordinates is more outstanding than that with the message timestamps. Comparing to the researches of traditional named entity recognition, the research makes full use of available information in LBS. In the future work, two issues need to be considered. The first one is the feature area zoning scheme. The second one is whether the performance of these features is obvious in a large dataset or does not need to be verified in future studies.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported in part by the Natural Science Foundation Programs of Shanghai Grants No. 13ZR1443100, No. 11ZR1440200, by ISTCP under Grant 2013DFM10100, by the National Science and Technology Support Plan Grant no. 2012BAH15F03, and by NSFC under Grants 51034003, 51174210.

References

[1] I. A. Junglas and R. T. Watson, "Location-based services," *Communications of the ACM*, vol. 51, no. 3, pp. 65–69, 2008.

[2] A. Chen, F. Peng, R. Shan, and G. Sun, "Chinese named entity recognition with conditional probabilistic models," in *Proceedings of the 5th SIGHAN Workshop on Chinese Language Processing*, 2006.

[3] Y. Wu, J. Zhao, B. Xu, and H. Yu, "Chinese named entity recognition based on multiple features," in *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pp. 427–434, October 2005.

[4] N. V. Sobhana, M. Pabitra, and S. K. Ghosh, "Conditional random field based named entity recognition in geological text," *International Journal of Computer Applications*, vol. 1, no. 3, pp. 143–147, 2010.

[5] Q. Qiu, D. Miao, and Z. Zhang, *Named Entity Recognition on Chinese Microblog, Computer Science*, 2013.

[6] X. Liu and M. Zhou, "Two-stage NER for tweets with clustering," *Information Processing & Management*, vol. 49, no. 1, pp. 264–273, 2013.

[7] G. Zhou and J. Su, "Named entity recognition using an HMM-based chunk tagger," in *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, 2002.

[8] A. McCallum, D. Freitag, and F. C. N. Pereira, "Maximum entropy Markov models for information extraction and segmentation," in *Proceedings of the 70th International Conference on Machine Learning (ICML '00)*, 2000.

[9] K.-J. Lee, Y.-S. Hwang, S. Kim, and H.-C. Rim, "Biomedical named entity recognition using two-phase model based on SVMs," *Journal of Biomedical Informatics*, vol. 37, no. 6, pp. 436–447, 2004.

[10] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional random fields: probabilistic models for segmenting and labeling sequence data," in *Proceedings of the 80th International Conference on Machine Learning (ICML '01)*, pp. 282–289, 2001.

[11] S. Shalev-Shwartz, Y. Singer, and N. Srebro, "Pegasos: primal estimated sub-GrAdient sOlver for SVM," in *Proceedings of the 24th International Conference on Machine Learning (ICML '07)*, pp. 807–814, June 2007.

[12] M. Collins, "Discriminative training methods for hidden Markov models: theory and experiments with perceptron algorithms," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2002.

[13] D. C. Liu and J. Nocedal, "On the limited memory BFGS method for large scale optimization," *Mathematical Programming B*, vol. 45, no. 3, pp. 503–528, 1989.

[14] N. Okazaki, CRFsuite, http://www.chokkan.org/software/crfsuite/.

[15] R. Zhang, G. Kikui, and E. Sumita, "Sub word-based tagging by conditional random fields for Chinese word segmentation," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, 2006.